1 **THE EMBODIMENTS OF THE INVENTION IN WHICH AN EXCLUSIVE PROPERTY**

2 **OR PRIVILEGE IS CLAIMED ARE DEFINED AS THE FOLLOWS:**

3 1. A method for providing dynamic interaction between a pair of application programs by a

4 platform neutral interface of a terminal, the pair of applications including a requestor application

5 desiring access to a target application, the method comprising the steps of:

6 registering access information of the target application, the access information including

7 published access information made available in a data structure for retrieval by the platform

8 neutral interface;

9 receiving an access request by the platform neutral interface from the requestor

10 application, the access request including request content corresponding to the published access

11 information of the target application;

12 obtaining an interface component by using the request content to search the data

13 structure, the interface component configured for enabling communication between the platform

14 neutral interface and the target application in an access format expected by the target application;

15 and

16 employing the interface component by the platform neutral interface to satisfy the access

17 request of the requestor application for interaction with the target application.

18

19 2. The method according to claim 1, wherein the target application is selected from the

20 group comprising: the target application configured for communication in a compatible language

21 with the platform neutral interface; and the target application configured for communication in a

22 incompatible language with the platform neutral interface.

23

1   3.   The method according to claim 2, wherein the incompatible language is that used by a

2   native runtime environment of the terminal.

3

4   4.   The method according to claim 2, wherein the interface component is an application

5   program interface (API) expressed in the compatible language.

6

7   5.   The method according to claim 2, wherein the interface component is an extension

8   element configured for providing mediation between the platform neutral interface and the target

9   application expressed in the incompatible language.

10

11   6.   The method according to claim 5 further comprising the step of registering the extension

12   element with the platform neutral interface through an extension interface, the published access

13   information of the extension element being added to the data structure.

14

15   7.   The method according to claim 6 further comprising the step of accessing the target

16   application through the platform neutral interface using the extension element to call a

17   corresponding application program interface (API) expressed in the incompatible language of the

18   target application.

19

20   8.   The method according to claim 2 further comprising the step of employing a search

21   algorithm with the request content for identifying matching ones of the interface component for

22   use by the platform neutral interface.

23

1   9.   The method according to claim 8, wherein the language used to express the platform

2   neutral interface is selected from the group comprising; a structured definition language and a

3   script.

4

5   10.   The method according to claim 9, wherein the structured definition language is based on

6   XML.

7
8   11.   The method according to claim 9, wherein the language used to express the script is

9   ECMAscript.

10

11 12.   The method according to claim 2 further comprising the step of assembling the request

12 content to include selected from the group comprising; a local location and a remote location.

13

14 13.   The method according to claim 12, wherein the remote location is on another terminal

15 coupled to said terminal through a network, the other terminal having one of the pair of

16 applications for network interaction with the other of the pair of applications.

17

18 14.   The method according to claim 13, wherein said terminal is configured as a client of a

19 schema defined service accessible over the network.

20

21 15.   The method according to claim 2, wherein the data structure is selected from the group

22 comprising an application profile table and an application API descriptor table.

23

1   16.   The method according to claim 15, wherein the application profile table includes

2   application profiles of a plurality of target applications.

3

4   17.   The method according to claim 15, wherein the application API descriptor table includes

5   descriptors selected from the group comprising; API descriptors and extension element

6   descriptors.

7
8   18.   The method according to claim 15, wherein the data structure includes the access

9   information selected from the group comprising; application URI, application version,

10   application description, and a predefined set of matching API construct pairs.

11

12   19.   The method according to claim 2 further comprising the step of providing an interface of

13   the platform neutral interface selected from the group comprising; an extension interface, a query

14   and registration interface, and an execution interface.

15

16   20.   The method according to claim 19, wherein the extension interface is configured for

17   dynamically extending a coupling of a new said interface component to the platform neutral

18   interface.

19

20   21.   A terminal for providing dynamic interaction between a pair of application programs in a

21   platform neutral environment provided by the runtime environment of the terminal, the pair of

22   applications including a requestor application desiring access to a target application, the terminal

23   comprising:

24   a data structure for registering access information of the target application, the access

25   information including published access information;

1   an interface module for providing the platform neutral environment, the interface module

2 configured for receiving an access request from the requestor application, the access request

3 configured to include request content corresponding to the published access information of the

4 target application, the published access information of the data structure retrievable by the

5 interface module; and

6   an interface component coupled to the interface module retrievable by using the request

7 content to search the data structure, the interface component configured for enabling

8 communication between the interface module and the target application in an access format

9 expected by the target application;

10   wherein employing the interface component by the interface module satisfies the access

11 request of the requestor application in interaction with the target application.

12

13 22. The terminal according to claim 21, wherein the target application is selected from the

14 group comprising: the target application configured for communication in a compatible language

15 with the interface module; and the target application configured for communication in a

16 incompatible language with the interface module.

17

18 23. The terminal according to claim 22, wherein the incompatible language is that used by

19 the native runtime environment of the terminal.

20

21 24. The terminal according to claim 22, wherein the interface component is an application

22 program interface (API) expressed in the compatible language.

23

1   25.   The terminal according to claim 22, wherein the interface component is an extension

2   element configured for providing mediation between the interface module and the target

3   application expressed in the incompatible language.

4

5   26.   The terminal according to claim 25 further comprising an extension interface for

6   registering the extension element with the interface module, the published access information of

7   the extension element configured for adding to the data structure.

8

9   27.   The terminal according to claim 26 further comprising a corresponding application

10   program interface (API) callable by the extension element step for accessing the target

11   application through the interface module, the application program interface (API) expressed in

12   the incompatible language of the target application.

13

14   28.   The terminal according to claim 22 further comprising a search algorithm for using the

15   request content to identify matching ones of the interface component for use by the interface

16   module.

17

18   29.   The terminal according to claim 28, wherein the language used to express the interface

19   module is selected from the group comprising; a structured definition language and a script.

20

21   30.   The terminal according to claim 29, wherein the structured definition language is based

22   on XML.

23

1   31.   The terminal according to claim 29, wherein the language used to express the script is

2   ECMAscript.

3

4   32.   The terminal according to claim 22, wherein the request content is configured to include

5   selected from the group comprising; a local location and a remote location.

6

7   33.   The terminal according to claim 32, wherein the remote location is on another terminal

8   coupled to said terminal through a network, the other terminal having one of the pair of

9   applications for network interaction with the other of the pair of applications.

10

11   34.   The terminal according to claim 33, wherein said terminal is configured as a client of a

12   schema defined service accessible over the network.

13

14   35.   The terminal according to claim 22, wherein the data structure is selected from the group

15   comprising; an application profile table and an application API descriptor table.

16

17   36.   The terminal according to claim 35, wherein the application profile table includes

18   application profiles of a plurality of target applications.

19

20   37.   The terminal according to claim 35, wherein the application API descriptor table includes

21   descriptors selected from the group comprising; API descriptors and extension element

22   descriptors.

23

1   38.   The terminal according to claim 35, wherein the data structure includes the access

2   information selected from the group comprising; application URI, application version,

3   application description, and a predefined set of matching API construct pairs.

4

5   39.   The terminal according to claim 22 further comprising an interface of the interface

6   module selected from the group comprising; an extension interface, a query and registration

7   interface, and an execution interface.

8

9   40.   The terminal according to claim 39, wherein the extension interface is configured for

10   dynamically extending a coupling of a new said interface component to the interface module.

11

12   41.   The terminal according to claim 39, wherein the query and registration interface is

13   configured for publishing the access information related to the interface component.

14
15   42.   A computer program product for providing dynamic interaction between a pair of

16   application programs in a platform neutral environment provided by a runtime environment of a

17   terminal, the pair of applications including a requestor application desiring access to a target

18   application, the computer program product comprising:

19   a computer readable medium;

20   a data structure module stored on the computer readable medium for registering access

21   information of the target application, the access information including published access

22   information;

23   an interface module coupled to the data structure module for providing the platform

24   neutral environment, the interface module configured for receiving an access request from the

25   requestor application, the access request configured to include request content corresponding to

1   the published access information of the target application, the published access information of

2   the data structure module retrievable by the interface module; and

3   an interface component module coupled to the interface module, the interface module

4   configured for containing an interface element retrievable by using the request content to search

5   the data structure module, the interface component configured for enabling communication

6   between the interface module and the target application in an access format expected by the

7   target application;

8   wherein employing the interface component by the interface module satisfies the access

9   request of the requestor application in interaction with the target application.